How to Become a Software Architect

Co-authored by Stephen Cognetta, MBA, Janice Tieperman , and 2 contributors
Last Updated: April 8, 2021

Similar to software engineers, software architects help to plan, develop, test, and provide various organizations with functional IT systems.[1] Before applying for a job as a software architect, spend some time studying computer science and programming so you can understand the basics of developing and managing a software system. Once you have a professional degree and certification, work in a team environment to get your bearings in the field. With enough programming and active practice under your belt, you'll be ready to put your best foot forward as a software architect!

Method 1 of 3: Getting the Proper Education and Certification

1 **Receive a bachelor's degree in computer science.** Enroll in an undergraduate program where you can learn computer programming in-depth. To get started, look at your nearest college or university to see what kinds of programs they offer. If you can't attend school in-person, see if you can enroll in an online program.[2]

- Programs with a focus on IT are also good options to consider.
- If you aren't a fan of programming or learning different coding languages, then software architecture might not be the career for you.

2 **Take the right coursework to become fluent in a coding language.** If you don't have a professional background in different programming languages, search online for free information and courses. Focus on the languages that you're likely to come across in your career, so you can become fluent. For specific lessons, check the individual websites of different programming languages.[3] In an active learning environment like a classroom, you might have an easier time learning to read code before you start writing it.[4]

- Ruby, otherwise known as Ruby on Rails, is a simple code used for many different web applications.[5]
- Java is used in a variety of mediums, like phone apps and video games, while JavaScript is used to add multimedia elements to different sites.
- Python is used in a lot of interactive web applications, like Instagram and Pinterest.

3 **Enroll in online courses for specific software styles.** Search online to find books, courses, and other informational sources on specific software interfaces and patterns.[6] Focus on learning and understanding the most basic patterns, such as layered, client-server, pipe-filter, and event-bus. If you're hoping to land a career in a specific field, study the software patterns that you're most likely to run into.[7]

- For instance, Android software development uses the event-bus pattern, while many speech recognition programs use the blackboard pattern.
- Sites like Udacity offer courses on different software/communication styles.
- Some undergraduate computer science programs discuss will discuss different kinds of software styles.[8]

4 **Complete an internship with a software company.** Search on different job hunting sites for open internship opportunities in the software architecture field. Start looking for jobs in August or September, focusing specifically on positions that offer some kind of salary. If you have certain career goal in mind, search for openings at companies that you could see yourself working at later on.[9]

- If you're applying as a college student, ask your professor for internship recommendations.

5 **Pursue a master’s degree to advance your qualifications.** Set yourself apart for more distinguished, high-paying jobs by getting a graduate degree in software engineering, or a similar field. Choose a program that prepares you for leadership positions, and teaches you how to manage different projects. Additionally, select a program that teaches you a lot about risk management and technical planning, so you’ll feel more confident in tackling large software projects.[10]

- Technological institutes offer specific graduate programs that may help you advance your career.

6 **Test for certification in specific coding fields.** Search online for different certification exams that you can take for different programming languages. Use the resources provided on the official website to test your knowledge at an entry-level, intermediate, or professional level. To qualify for more high-level jobs in the software architecture field, complete and pass the most advanced exam offered.[11]

- You need to gain lower certifications before applying for an advanced certificate.
- Some exams need to be taken in person, like the Associate- and Professional-Level Python certification exams.

Method 2

Method 2 of 3:
Developing Experience in the Field

1 **Work as a software developer in a collaborative environment.** Before managing and testing a software system, look for lower level jobs where you can work as part of a software development team. Look on a job search sites for different software developer positions, and see if any of them match your particular skill set. As you search for different positions, look for jobs that will expand your horizons while also pushing you out of your comfort zone.[12]

- The architecture field involves designing, testing, and maintaining different kinds of software. If you're more familiar with testing and maintaining a program, look for a position that focuses more on testing.

2 **Create a website or application where you can practice coding.** Use a free, open source website to practice and test out different codes that you've been working on. If you'd like to get more proficient in a certain programming language, design a practice website or application that utilizes this language. As you hone your skills, try making your digital project complex with more advanced coding.[13] [14]

- GitHub or Java is a great place to write, test, and practice different types of code.

3 **Network with other architects through conferences.** Look online to find a technical, software-centric conference near you. While visiting this event, attend lectures that cater specifically to software architecture. While at the conference, try introducing yourself to other attendees and speakers; with any luck, you might make some new, professional connections! [15]

- If you work as a software developer or architect, ask your employer if they'd be willing to sponsor your trip to the conference!

4 **Ask an experienced coworker to mentor you.** Don’t be discouraged if you’re struggling with certain aspects of your job, like coding, testing, or decision-making. Instead, reach out to a senior coworker and ask them to walk you through an especially difficult task.[16] Additionally, ask if you can see some of their code, so you have a better idea of what to write for future projects.[17]

- If you can’t find a mentor at work, check on social media sites and blogs for software professionals with advice to give.

5 **Teach beginners in the field to gain more confidence in your skills.** You don’t have to be a professor or teacher to share your knowledge with others! Instead, use a blog or online forum to post some information about a specific software pattern or programming language. As you post more, you might make valuable connections with others along the way![18]

- For instance, if you feel confident with the client-server software pattern, create a blog dedicated to that specific topic.
- Teaching others about different subjects can help boost your own confidence levels!

Method 3

Method 3 of 3: Finding Work as a Software Architect

1 **Develop a resume that shows your professional credentials.** When applying for an official position as a software architect, cater your resume and cover letter to your potential employer. Highlight your ability to plan ahead and solve different software issues as they arise. Mention the different high-pressure positions and projects that you've worked on in the past, and how these experiences prepared you for a job in software architecture.[19]

- Take note of your programming language skills and any certifications that you have.
- If you have a master's degree, highlight the advanced skills that you learned in the program.

2 **Apply for work as a release engineer to help companies integrate new software.** If you like thinking on your feet, look for release engineer job openings. In this type of job, your skills as a software architect will be focused on software testing and maintenance, and ensuring that a software program is completely functional. Additionally, you'll have to develop different tools, like build automation and project metrics.[20]

- Release engineers aim to create builds that don't require new releases.

3 **Work as an applications architect if you want to work with specific programs.** If you enjoy working with more individualized programs like games and word processors, apply for a job as an applications or software architect. Depending on the position, you might be working in a large IT organization; however, your focus will be on the development, testing, and maintenance of smaller pieces of software.[21]

- For instance, some applications architects might work with cloud-based software.

4 **Find a job as a systems software developer if you prefer to work with operating systems.** Search online for a career in system software development, where you focus on developing, testing, and maintaining software interfaces. Depending on the job, you could work for computer manufacturing company or a more specific company. Choose a job that best fits your particular niche in the field of software architecture.[22]

- For instance, if you're familiar with the client-server software pattern, you might be well-equipped to develop and test software for a bank.[23]

> **Did you know?** The term "developer" can be used interchangeably with "architect" when it comes to software-related careers.[24]

5 **Study specific industries if you want to take a specialized job.** While many open positions for software architects are with computer and tech companies, note that there are countless independent groups that need guidance and assistance with their software systems. Before applying for a job in a specific industry, research the exact technological needs of that position. Take some time to brush up on the preferred programming language of a certain career path before seeking out job opportunities.[25]

- For instance, if you're working for a large corporation like Facebook or Wordpress, you'll need to be familiar with PHP, or Hypertext Preprocessor. If you're working with Microsoft-specific programs, you'll need to be familiar with C#.
- Try to stay on top of which programming languages are the most relevant. For example, while C is a useful programming language, it isn't as practical or well-used as its descendants, Java and C#.

Expert Q&A

Question

What kind of interview questions will I have about software architecture?

Stephen Cognetta, MBA
Tech Interviewing Coach
Expert Answer

You can expect a lot of algorithm questions. You may be asked to design a heuristic or a solution for a given problem, and explain your thinking behind the design. You can also expect some open-ended questions as well. Things like, "How would you go about designing Instagram?" and questions like that.

Helpful 0 Not Helpful 0

Question

How can I prepare for a software architect interview?

Stephen Cognetta, MBA
Tech Interviewing Coach
Expert Answer

Exponent is my overall suggestion for practice. The more you can practice, the better. There are also a lot of good books out there for practice, like Cracking the Coding Interview. LeetCode is another popular tool. Beyond that, just prepare for the super obvious questions.

Helpful 0 Not Helpful 0

References

1. ↑ https://www.prospects.ac.uk/jobs-and-work-experience/job-sectors/information-technology/how-to-become-a-technical-architect
2. ↑ https://www.computercareers.org/6-steps-to-become-a-software-architect/
3. ↑ https://www.computercareers.org/5-steps-to-become-a-computer-programmer/
4. ↑ http://depts.washington.edu/csed/vandegrift_research_plan.pdf
5. ↑ https://www.inc.com/larry-kim/10-most-popular-programming-languages-today.html
6. ↑ https://www.computercareers.org/6-steps-to-become-a-software-architect/
7. ↑ https://towardsdatascience.com/10-common-software-architectural-patterns-in-a-nutshell-a0b47a1e9013?gi=f8b39847958f
8. ↑ http://catalog.davidson.edu/preview_program.php?catoid=21&poid=1404&hl=computer%20science&returnto=search
9. ↑ https://www.freecodecamp.org/news/landing-a-software-engineering-internship-66e0d541539a/
10. ↑ https://www.stevens.edu/school-systems-enterprises/masters-degree-programs/software-engineering
11. ↑ https://pythoninstitute.org/certification/
12. ↑ https://www.computercareers.org/6-steps-to-become-a-software-architect/
13. ↑ https://www.computercareers.org/5-steps-to-become-a-computer-programmer/
14. ↑ Stephen Cognetta, MBA. Computer Engineer. Expert Interview. 21 July 2020.
15. ↑ https://www.sciencemag.org/careers/2017/05/how-get-most-out-attending-conferences
16. ↑ Stephen Cognetta, MBA. Computer Engineer. Expert Interview. 21 July 2020.
17. ↑ http://www.wicsa.net/how-to-become-a-software-architect/
18. ↑ http://www.wicsa.net/how-to-become-a-software-architect/
19. ↑ https://www.prospects.ac.uk/jobs-and-work-experience/job-sectors/information-technology/how-to-become-a-technical-architect
20. ↑ https://www.usenix.org/sites/default/files/conference/protected-files/lisa15_slides_mcnutt.pdf
21. ↑ https://www.prospects.ac.uk/jobs-and-work-experience/job-sectors/information-technology/how-to-become-a-technical-architect
22. ↑ https://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm
23. ↑ https://towardsdatascience.com/10-common-software-architectural-patterns-in-a-nutshell-a0b47a1e9013?gi=f8b39847958f
24. ↑ https://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm#tab-2
25. ↑ https://www.inc.com/larry-kim/10-most-popular-programming-languages-today.html

About This Article

Co-authored by:
Stephen Cognetta, MBA
Tech Interviewing Coach

This article was co-authored by Stephen Cognetta, MBA and by wikiHow staff writer, Janice Tieperman. Stephen Cognetta is the co-founder and CEO of Exponent, a learning platform that helps people prepare for and ace their tech interviews. Stephen specializes in coaching for product management, software engineering, product marketing, management, technical project management, and data science interviews. Stephen holds a BS in Computer Engineering from Princeton University, where he graduated Summa Cum Laude, and an MBA from Stanford University. Prior to founding Exponent, Stephen worked as a Product Manager for Google and co-founded HackMentalHealth. This article has been viewed 2,301 times.

Co-authors: **2**
Updated: **April 8, 2021**
Views: **2,301**

Categories: Occupations

https://www.wikihow.com/Become-a-Software-Architect